



- A venue operator implements the instant invention for his use as his own ticketing and reservation system.
- 2. That specific venue is recorded, coded and placed on a server connected conventionally to the internet or any such wide area network.
- 3. A prospective customer for tickets to any some event logs onto the internet, or other such wide area network, in his conventional manner, whether from home or office or any remote location.
- 4. Said prospective customer attaches to the server through any conventional graphical browsing means and views the available venues, performances, dates, and/or other such offerings.
- 6. Said prospective customer selects a specific area of interest from the available venues. Such may be theater tickets for a particular show date, airline seats for a given flight, etc., and indicates his selection through a conventional hyperlink or other compatible means.
- 5. Upon first access an inquiry is made to the appropriate database asking for a return to the prospective customer of all generalized information available from which he will make a selection of the type of venue he is seeking.
- 7. Upon contact the server makes another appropriate database query asking for a return to the prospective customer of all specific information relating to his selection, i.e., available seats for the chosen airline flight, from which he will make his selection or
- 8. Upon return the prospective customer is presented with a representation of all available seating for his selected venue. From this representation, which may be graphical or displayed in any other appropriate way, he makes a selection of the specific seat or seats he wishes to reserve and submits such to the server.
- 10. The server requests payment information from the now customer through a form input mechanism. Once completed and submitted customers payment information is verified through conventional means with any of the existing processing means.
- 9. The server creates a temporary customer identification and associates the prospective customers selections with that specific identification so as to preclude confusion and allow multiple simultaneous users.
- 11(A). If customer's payment information verification is successful then:
- (a) the customers information, as supplied in Step 10, is made permanent and;
- (b) those specific seats as selected by the customer are removed from available inventory and;
- (c) marked as unavailable on the graphical representation of the venue seating as presented to the next prospective customer;
- (d) entered into the accounting and administration information database for later retrieval by the venue operator..

- 11(B). If customers payment information verification is not successful then:
- (a) customer is given the opportunity to correct his submittal and try again;
- (b) customer can return to the beginning and repeat the entire process.
 - 12. A confirmation of the transaction, containing the transaction identification and other pertinent information is returned to the customer

Figure 1

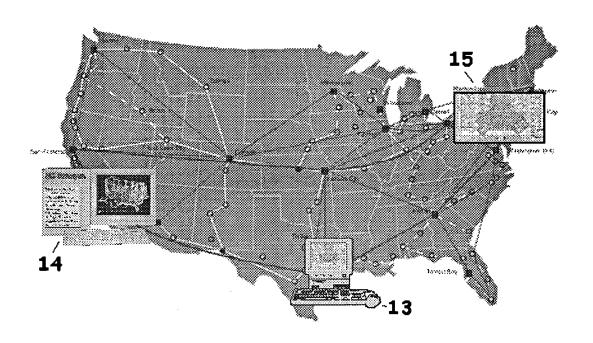


Figure 2

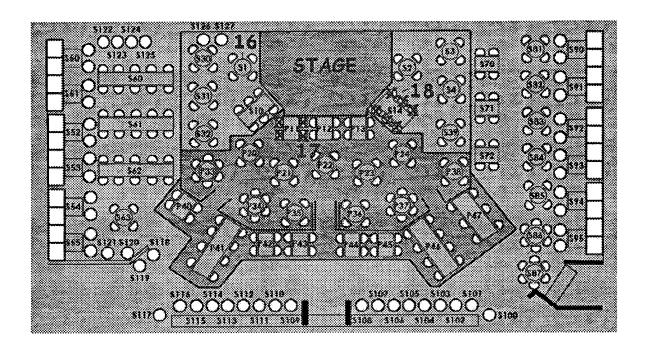


Figure 3a

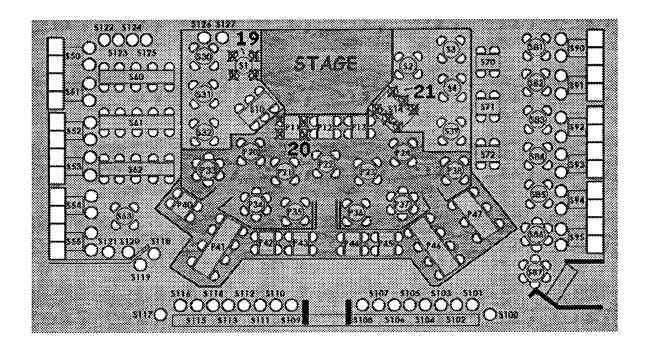
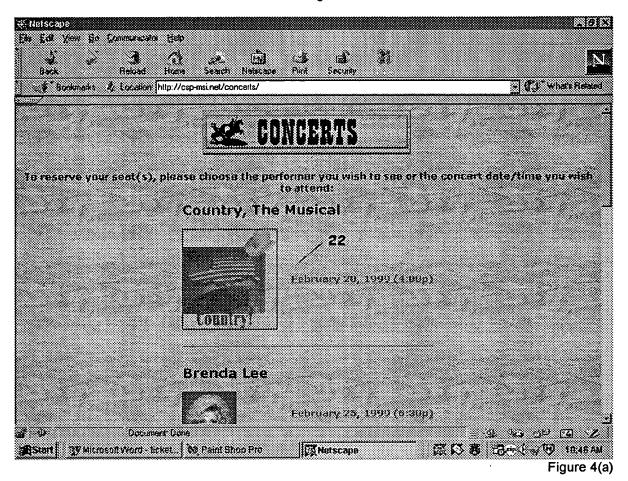


Figure 3b

Start Figure 4



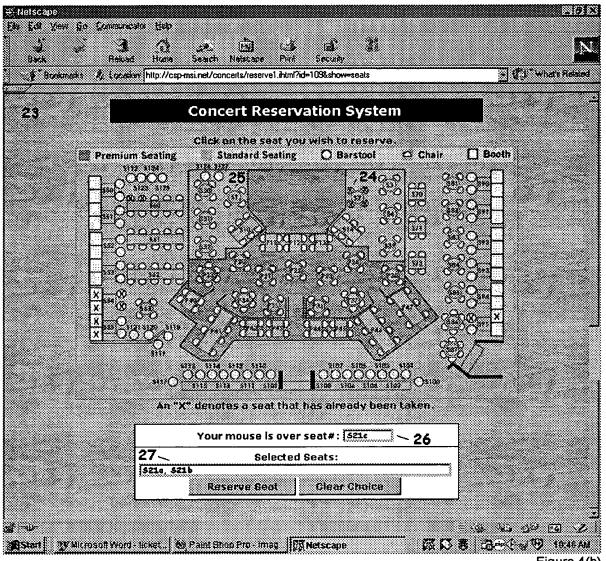


Figure 4(b)

•	oncert Reservation System	,
	CUSTOMER INFORMATION	
Name as appear on CC	Richard Halavais	
Phone	714-693-1171	
Email	rhal@msintergate.com	
CC Number	411111111111111	
Expiration (MM/YY)	Month 09 / Year 99 example: 12/99	
	Reserve Seats Clear Choices 29	

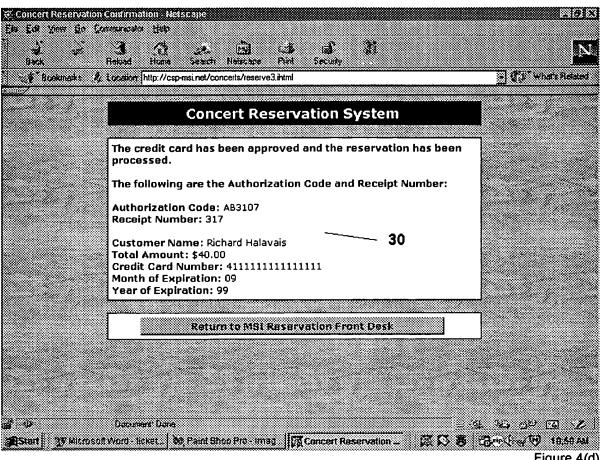
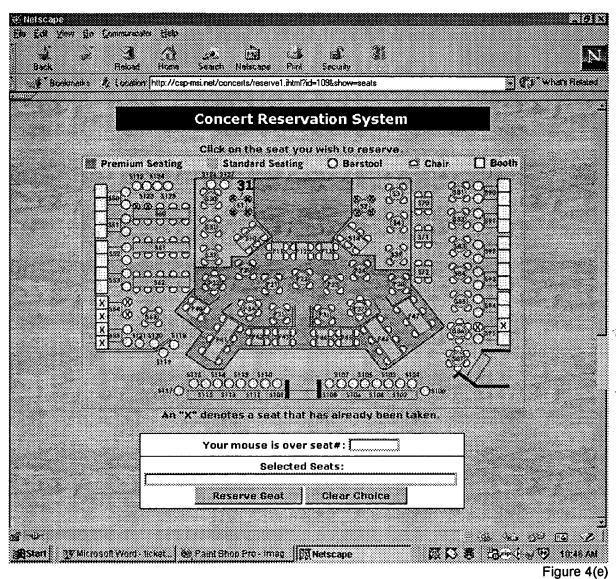


Figure 4(d)



i igule 4

End Figure 4

Start Figure 5

Begin Program

```
<Markup language>
<REM --- Imports the file "datasource.inc" which creates variable "datasource" which is used
to tell markup language datasource to connect to.
                                                              --- >
<INCLUDE NAME="database\datasource.inc">
<REM --- In case a database or other type of error occurs, this display the error message. --</p>
-- >
<ERROR>
<FONT FACE="Verdana, Arial" SIZE="+1"><B>An Error Has Occurred</b>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>Error Message = :i_errortext
</B></FONT><P>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>database Error =
:i_databaseerrortext</B></FONT><P>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>database Error = :i_databaseerrorstmt
</B></FONT><P>
</ERROR>
<REM --- Begin normal markup language here
<markup language>
<HEAD>
<TITLE>Ticketing & amp; Reservation System</TITLE>
</HEAD>
<BODY BACKGROUND="images/background.jpg" TEXT="#000000" LINK="#006666"
VLINK="#006666">
<CENTER>
<IMG SRC="images/masthead_concerts.gif" HEIGHT=60 WIDTH=280><P>
<FONT FACE='Verdana, Arial' SIZE=-1><B>To begin reserving your seat(s), please select
the concert date/time you wish to attend next to the performer you want to
see:</B></FONT>
```

<TABLE BORDER=0 CELLSPACING=5 CELLPADDING=5> <REM --- Begin database query to retrieve all performances that is <REM --- currently available. Will loop until all available <REM --- performing artists and their performances are listed. <REM --- Part of the return from the query are the links that will --- > <REM --- take you to the next step of the reservation. <database DBNAME=":datasource"</pre> database="SELECT id, name, picture, sequence FROM category WHERE active=1 AND parent=-1 **ORDER BY sequence"** ALIAS="concert"> <databaseFETCH ALIAS="concert"> <WHILE NOTALIAS=i_databaseempty> <TR> <TD COLSPAN=2> :concert_name </TD></TR><TR><TD VALIGN="top"> </TD><TD> <Markup language DBNAME=":datasource"</p> database="SELECT id, name, date, time FROM category WHERE active=1 AND parent=:concert_id ORDER BY date, time" OUTPUT=" :2 "> </TD></TR><TD></TD></TD></TR>

<databaseFETCH ALIAS="concert">

</WHILE ALIAS=i_databaseempty> </database ALIAS="concert"> </TABLE> </BODY> </markup language> processing then passes to: <Markup language> <REM --- Imports the file "datasource.inc" which creates the <REM --- variable "datasource" which is used to tell Markup language which --- > <REM --- database datasource to connect to. <INCLUDE NAME="database\datasource.inc"> <REM In case a database or other type of error occurs, this will display the error message. > <ERROR> An Error Has Occurred/FONT><P> Error Message = :i_errortext<P> database Error = :i_databaseerrortext<P> database Error = :i_databaseerrorstmt<P> </ERROR> <REM --- Begin normal markup language here --- > <markup language> <HEAD> <TITLE>Ticketing & amp; Reservation System - Select Seat(s)</TITLE> </HEAD> <REM --- All seats clicked will pass its information to a input box, "newseats". When done, the information will be passed to "process.ihtml" and be processed by "reserve2.ihtml". --- > <REM --- This code allows the ability to select multiple seats before proceeding to the next

step of reservation process. Other features include listing the selected seats in the

```
"number" text box to show the seats that has been clicked. This script also alters the button
text to be grammatically correct. --- >
<SCRIPT LANGUAGE="JavaScript">
<!--
function selectseat(idnum, seatnum)
{
reserved = document.seats.number.value;
if (reserved == "")
{
document.seats.newseats.value = "<Markup language DBNAME=:"+"datasource
database='INSERT INTO basket (custid, pid, qty) VALUES (:"+"custid, "+idnum+",
1)'><database DBNAME=:"+"datasource database='SELECT cost FROM products WHERE
id="+idnum+"'><databaseFETCH><iEQ NAME='ticketprice'
VALUE=:"+"1></database><iEQ NAME='total' VALUE=<iEVAL EXPR=':"+"ticketprice +
:"+"total' PREC='2'>><Markup language DBNAME=:"+"datasource database='INSERT INTO
orderdetail (pid, oid, qty, sell) VALUES ("+idnum+", :"+"oid, 1, :"+"ticketprice)'>";
document.seats.number.value = seatnum;
document.seats.button1.value = "Reserve Seat";
document.seats.button2.value = "Clear Choice";
}
else
document.seats.newseats.value += "<Markup language DBNAME=:"+"datasource
database='INSERT INTO basket (custid, pid, qty) VALUES (:"+"custid, "+idnum+",
1)'><database DBNAME=:"+"datasource database='SELECT cost FROM products WHERE
id="+idnum+""><databaseFETCH><iEQ NAME='ticketprice'
VALUE=:"+"1></database><iEQ NAME='total' VALUE=<iEVAL EXPR=':"+"ticketprice +
"+"total' PREC='2'>><Markup language DBNAME=:"+"datasource database='INSERT INTO
orderdetail (pid, oid, qty, sell) VALUES ("+idnum+", :"+"oid, 1, :"+"ticketprice)'>";
document.seats.number.value = reserved+", "+seatnum;
```

```
document.seats.button1.value = "Reserve Seats";
document.seats.button2.value = "Clear Choices";
}
}
function clear()
{
document.seats.newseats.value = "";
}
//-->
</SCRIPT>
<BODY BACKGROUND="images/background.jpg" TEXT="#000000" LINK="#006666"</p>
VLINK="#006666">
<REM --- Begin database query to retrieve the last id number from table "orderdetail" for the
purpose of creating a new instance of the stage image whenever a new order has occurred.
This will prevent the browser from displaying an old floor image from its cache as a different
image name is called every time the file is run. --- >
<database DBNAME=":datasource"</pre>
       database="SELECT max(id)
       FROM orderdetail">
       <databaseFETCH>
       <iEQ NAME="imagenum" VALUE=:1>
       <databaseFETCH>
</database>
<iEQ NAME="imagetype" VALUE=".jpg">
<CENTER>
<TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=5 CELLSPACING=0
WIDTH=500>
<TR><TD><CENTER>
<FONT FACE='Verdana, Arial' SIZE=+1 COLOR='#FFFFFF'><B>
Concert Reservation System
```

```
</B></FONT>
</CENTER>
</TD></TR></TABLE></CENTER>
<REM - Loads the core image that will be dynamically altered for use as the image map. - >
<iIMAGEFROMFILE NAME="stage" FILENAME="images/stage.jpg" TYPE="jpeg">
<MAP NAME="stage" BORDER=0>
<REM --- Begin database query to retrieve all seat information for the image map "stage".
Will loop until all available seat information for this particular performance is listed.
<REM --- The return query will be used only if the "active" attribute of that particular seat is
marked "true" or "1". --- >
<REM --- If active, the seat's information, including image map coordinates will be
displayed. Otherwise, the seat on the image map will be x'd out according to the x and y
coordinates associated with it.
                                             --- >
<database DBNAME=":datasource"</pre>
database="SELECT id, active, x, y, shape, corrds, name, mouseout, mouseover1,
mouseover2, mouseover3
       FROM products
       WHERE catid=:id">
       <databaseFETCH>
       <iWHILE NOTALIAS=i_databaseempty>
       <iEQ NAME="active" VALUE=:2>
       <iCASE ALIAS="active" VALUE=1>
<AREA SHAPE=":5" COORDS=":6" HREF="javascript:selectseat(:1,':7')" ALT="Seat # :7"</pre>
OnMouseOut=:8 OnMouseOver=:9:7:10:7:11 >
       </iCASE ALIAS="active">
       <iCASE ALIAS="active" VALUE=0>
       <iIMAGETEXT NAME="stage" TEXT="X" X=:3 Y=:4 COLOR="red">
       </iCASE ALIAS="active">
       <databaseFETCH>
       </iWHILE ALIAS=i_databaseempty>
```

```
</database>
</MAP>
<CENTER>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>
Click on the seat you wish to reserve.
</B></FONT>
<TABLE BORDER=1 CELLPADDING=0 CELLSPACING=0>
<TR><TD>
<REM --- Converts the dynamically generated image, "stage", into a web friendly image
type - "jpeg".
<iIMAGEWRITE NAME="stage" FILENAME="images/ch_stage-:id-:imagenum:imagetype"</p>
TYPE="ipeg" QUALITY="30">
<REM --- Deletes the temporary image "stage" as it is no longer needed. --- >
<iIMAGEDESTROY NAME="stage">
<IMG SRC="images/legend.jpg" BORDER=0 HEIGHT=20 WIDTH=584><BR>
<IMG SRC="images/ch_stage-:id-:imagenum:imagetype" BORDER=0 USEMAP="#stage"</pre>
HEIGHT=315 WIDTH=584>
</CENTER>
</TD></TR></TABLE>
<FONT FACE='Verdana, Arial' SIZE=-1><B>
An "X" denotes a seat that has already been taken.
</B></FONT>
<FORM NAME="seats" ACTION="process.ihtml" METHOD="post">
<CENTER>
<TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR><TD>
<TABLE BGCOLOR='#FFFFFF' BORDER=0 CELLPADDING=5 CELLSPACING=1>
<TR><TD><CENTER>
<FONT FACE='Verdana, Arial' SIZE=-1><B>
```

Your mouse is over seat#: <INPUT TYPE="text" NAME="seatnum" SIZE=5>

```
</B></FONT></CENTER>
       </TD></TR><TR><TD><CENTER>
       <FONT FACE='Verdana, Arial' SIZE=-1><B>
       Selected Seats: <BR><INPUT TYPE="text" NAME="number" SIZE=40>
       <BR>
       <INPUT TYPE="hidden" NAME="newseats" SIZE=50>
       <INPUT TYPE="submit" NAME="button1" VALUE="Reserve Seat">&nbsp;<INPUT</p>
       TYPE="reset" NAME="button2" VALUE="Clear Choice" onClick="clear()">
       </B></FONT>
       </CENTER>
       </TD></TR></TABLE></TD></TR></CENTER>
       </FORM></CENTER></BODY>
       </markup language>
which then passes to a template:
       <Markup language>
       <REM --- Imports the file "datasource.inc" which creates the variable "datasource" which is
       used to tell Markup language which ODBC datasource to connect to.
       <iINCLUDE NAME="database\datasource.inc">
       <REM --- In case a database or other type of error occurs, this will display the error
       message. - >
       <ERROR>
       <FONT FACE="Verdana, Arial" SIZE="+1"><B>An Error Has Occurred</B></FONT><P>
       <FONT FACE="Verdana,Arial" SIZE="-1"><B>Error Message =
       :i_errortext</B></FONT><P>
       <FONT FACE="Verdana, Arial" SIZE="-1"><B>database Error =
       :i_databaseerrortext</B></FONT><P>
       <FONT FACE="Verdana, Arial" SIZE="-1"><B>database Error =
       :i_databaseerrorstmt</B></FONT><P>
       </ERROR>
```

```
<REM --- Copy a preformatted file, "empty.intml" into a new file, --- >
       <REM --- "reserve2.ihtml".
       <iCOPYFILE SRC="empty.ihtml" DST="reserve2.ihtml">
       <REM --- Appends the information passed from "reseve1.ihtml" to "reserve2.ihtml" for
       process. Other html information is also passed.
                                                                                --- >
       <iFILE NAME="reserve2.ihtml" DATA=":newseats" OP="append">
       <iFILE NAME="reserve2.ihtml" DATA="<INPUT TYPE='hidden' NAME='total'
       VALUE=:total>" OP="append">
       <iFILE NAME="reserve2.ihtml" DATA="</FORM>" OP="append">
       <iFILE NAME="reserve2.ihtml" DATA="</CENTER>" OP="append">
       <iFILE NAME="reserve2.ihtml" DATA="<P>" OP="append">
       <iFILE NAME="reserve2.ihtml" DATA="</BODY>" OP="append">
       <iFILE NAME="reserve2.ihtml" DATA="</markup language>" OP="append">
       <REM --- Once the information is appended into "reserve2.ihtml", it will be automatically
       push the page "reserve2.ihtml" to the browser.
       <iREDIR URL="reserve2.ihtml">
which then combines information and passes to:
       <Markup language>
       <REM --- Imports the file "datasource.inc" which creates the variable "datasource" which is
       used to tell Markup language which ODBC datasource to connect to.
       <iINCLUDE NAME="database\datasource.inc">
       <REM --- In case a database or other type of error occurs, this will display the error
       message. - >
       <ERROR>
       <FONT FACE="Verdana, Arial" SIZE="+1"><B>An Error Has Occurred</B></FONT><P>
       <FONT FACE="Verdana,Arial" SIZE="-1"><B>Error Message =
       :i_errortext</B></FONT><P>
       <FONT FACE="Verdana, Arial" SIZE="-1"><B>database Error =
       :i_databaseerrortext</B></FONT><P>
```

```
X
<FONT FACE="Verdana, Arial" SIZE="-1"><B>database Error =
:i databaseerrorstmt</B></FONT><P>
</ERROR>
<REM --- Copy a preformatted file, "empty.ihtml" into a new file, "reserve2.ihtml". --- >
<iCOPYFILE SRC="empty.ihtml" DST="reserve2.ihtml">
<REM --- Appends the information passed from "reseve1.ihtml" to "reserve2.ihtml" for
process. Other html information is also passed. --- >
<iFILE NAME="reserve2.ihtml" DATA=":newseats" OP="append">
<iFILE NAME="reserve2.ihtml" DATA="<INPUT TYPE='hidden' NAME='total'
VALUE=:total>" OP="append">
<iFILE NAME="reserve2.ihtml" DATA="</FORM>" OP="append">
<iFILE NAME="reserve2.ihtml" DATA="</CENTER>" OP="append">
<iFILE NAME="reserve2.ihtml" DATA="<P>" OP="append">
<iFILE NAME="reserve2.ihtml" DATA="</BODY>" OP="append">
<iFILE NAME="reserve2.ihtml" DATA="</markup language>" OP="append">
<REM --- Once the information is appended into "reserve2.ihtml", it will be automatically
push the page "reserve2.ihtml" to the browser.
<iREDIR URL="reserve2.ihtml">
then (takes all information from index & reserve one and conbines it to with the information
in empty) then passes to:
<!Markup language>
<REM --- Imports the file "datasource.inc" which creates the variable "datasource" which is
used to tell Markup language which ODBC datasource to connect to. --- >
<iINCLUDE NAME="database\datasource.inc">
<REM -- In case a database or other type of error occurs, this will display the error message.
-- >
<ERROR>
<FONT FACE="Verdana, Arial" SIZE="+1"><B>An Error Has Occurred</B></FONT><P>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>Error Message =
:i errortext</B></FONT><P>
```

XI database Error = :i_databaseerrortext<P> database Error = :i_databaseerrorstmt<P> </ERROR> <REM --- Begin normal markup language here <markup language> <HEAD> <TITLE>Ticketing & District Control of the Control </HEAD> <BODY BACKGROUND="images/background.jpg" TEXT="#000000" LINK="#006666" VLINK="#006666"> <REM ---- Set Customer Info ---- > <CENTER> <TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=5 CELLSPACING=0 WIDTH=500> <TR><TD><CENTER> Ticketing & Programme Reservation System </CENTER> </TD></TR> </TABLE> <FORM ACTION=reserve3.ihtml METHOD=POST> <TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=0 CELLSPACING=0> <TR><TD> <TABLE BGCOLOR='#FFFFFF' BORDER=0 CELLPADDING=5 CELLSPACING=1> <TR> <TD COLSPAN=2><CENTER>CUSTOMER INFORMATION</CENTER></TD>

```
</TR><TR>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B>Name as appear on
CC</B></FONT></TD>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B><INPUT TYPE="text"
NAME="name" MAXLENGTH=50 SIZE=40></B></FONT></TD>
      </TR><TR>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B>Phone</B></FONT></TD>
      <TD><FONT FACE='Verdana.Arial' SIZE=-1><B><INPUT TYPE="text"
NAME="phone" MAXLENGTH=50 SIZE=12></B></FONT></TD>
      </TR><TR>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B>Email</B></FONT></TD>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B><INPUT TYPE="text"
NAME="email" MAXLENGTH=50 SIZE=40></B></FONT></TD>
      </TR><TR>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B>CC Number</B></FONT></TD>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B><INPUT TYPE="text"
NAME="ccnum" MAXLENGTH=50 SIZE=40></B></FONT></TD>
      </TR><TR>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B>Expiration
(MM/YY)</B></FONT></TD>
      <TD><FONT FACE='Verdana, Arial' SIZE=-1><B>Month <INPUT TYPE="text"
NAME="ccmexp" MAXLENGTH=50 SIZE=2> / Year <INPUT TYPE="text" NAME="ccyexp"
MAXLENGTH=50 SIZE=2> ... example: 12/99</B></FONT></TD>
      </TR><TR>
      <TD COLSPAN=2>
      <CENTER>
      <FONT FACE='Verdana, Arial' SIZE=-1><B>
      <INPUT TYPE="submit" VALUE="Reserve Seats">
      <BR>
      <INPUT TYPE="reset" VALUE="Clear Choices">
```

```
</B></FONT>
       </CENTER>
       </TD></TR>
</TABLE>
</TD></TR></TABLE>
<iEQ NAME="total" VALUE=0>
<iEQ NAME="date" VALUE='<iDATE>'>
<REM --- Begin database command to insert a new customer profile into the database. This
step is primarily for the purpose of obtaining a new customer id to associate this transaction.
The customer's ip and captured and inserted into a new record in the "customers" table.
The marker, "new" is flagged "true" for the attrieval of the new id. Once the new customer
id is captured, the marker "new" is turned off.
<Markup language DBNAME=:datasource</p>
       database="INSERT INTO customers (ip, new)
       VALUES (':i_ip',1)">
<database DBNAME=:datasource</pre>
       database="SELECT id
       FROM customers
       WHERE new=1">
       <databaseFETCH>
       <iEQ NAME="custid" VALUE=:1>
</database>
<Markup language DBNAME=:datasource</p>
       database="UPDATE customers
       SET new=0
       WHERE id=:custid">
<REM --- Begin database command to insert a new reservation into the database. This step
```

is primarily for the purpose of creating a new reservation record, flagged with the newly obtained customer id, so that the transaction information may be processed in

[&]quot;resere3.ihtml".

<Markup language DBNAME=:datasource</p>

database="INSERT INTO orders (orderdate, approvalcode, receiptnum, totalcharge, custid)

VALUES (':date','0000', '0000', :total, :custid)"

FAILURE="The order could not be processed at this time due to technical difficulties.">

<REM --- Begin database command to obtain the reservation id that was just created. This information, coupled with the customer id, will be used to identify this particular transaction in the database and be updated in "resere3.ihtml" --->

<database DBNAME=:datasource</pre>

database="SELECT max(id)

FROM orders">

<databaseFETCH>

<iEQ NAME="oid" VALUE=:1>

</database>

<INPUT TYPE="hidden" NAME="reserve" VALUE="receipt">

<INPUT TYPE="hidden" NAME=custid VALUE=:custid>

<INPUT TYPE="hidden" NAME=phone VALUE=:phone>

<INPUT TYPE="hidden" NAME=email VALUE=:email>

<INPUT TYPE="hidden" NAME=oid VALUE=:oid>

<Markup language DBNAME=:datasource database='INSERT INTO basket (custid, pid, qty)</p>
VALUES (:custid, 6431, 1)'><database DBNAME=:datasource database='SELECT cost</p>
FROM products WHERE id=6431'><databaseFETCH><iEQ NAME='ticketprice'</p>
VALUE=:1></database><iEQ NAME='total' VALUE=<iEVAL EXPR=':ticketprice + :total'</p>
PREC='2'>><Markup language DBNAME=:datasource database='INSERT INTO orderdetail</p>
(pid, oid, qty, sell) VALUES (6431, :oid, 1, :ticketprice)'><Markup language</p>
DBNAME=:datasource database='INSERT INTO basket (custid, pid, qty) VALUES (:custid, 6428, 1)'><database DBNAME=:datasource database='SELECT cost FROM products</p>
WHERE id=6428'><databaseFETCH><iEQ NAME='ticketprice'</p>

VALUE=:1></database><iEQ NAME='total' VALUE=<iEVAL EXPR=':ticketprice + :total'

PREC='2'>><Markup language DBNAME=:datasource database='INSERT INTO orderdetail



(pid, oid, qty, sell) VALUES (6428, :oid, 1, :ticketprice)'><Markup language

DBNAME=:datasource database='INSERT INTO basket (custid, pid, qty) VALUES (:custid, 6429, 1)'><database DBNAME=:datasource database='SELECT cost FROM products

WHERE id=6429'><databaseFETCH><iEQ NAME='ticketprice'

VALUE=:1></database><iEQ NAME='total' VALUE=<iEVAL EXPR=':ticketprice + :total'

PREC='2'>><Markup language DBNAME=:datasource database='INSERT INTO orderdetail

(pid, oid, qty, sell) VALUES (6429, :oid, 1, :ticketprice)'><Markup language

DBNAME=:datasource database='INSERT INTO basket (custid, pid, qty) VALUES (:custid, 6430, 1)'><database DBNAME=:datasource database='SELECT cost FROM products

WHERE id=6430'><databaseFETCH><iEQ NAME='ticketprice'

VALUE=:1></database><iEQ NAME='total' VALUE=<iEVAL EXPR=':ticketprice + :total'

PREC='2'>><Markup language DBNAME=:datasource database='INSERT INTO orderdetail

(pid, oid, qty, sell) VALUES (6430, :oid, 1, :ticketprice)'><INPUT TYPE='hidden'

NAME='total' VALUE=:total></FORM></CENTER><P></BODY></markup language>

takes in credit card information and passes to:

<!Markup language>

<REM --- Imports the file "datasource.inc" which creates the variable "datasource" which is used to tell Markup language which ODBC datasource to connect to.

<iINCLUDE NAME="database\datasource.inc">

<REM -- In case a database or other type of error occurs, this will display the error message.

-- >

<ERROR>

An Error Has Occurred

Error Message =

:i_errortext<P>

database Error =

:i_databaseerrortext<P>

database Error =

:i databaseerrorstmt<P>

</ERROR>

XVI <REM --- Begin normal markup language here <markup language> <HEAD> <TITLE>Ticketing & Direction System - Confirmation</TITLE> </HEAD> <BODY BACKGROUND="images/background.jpg" TEXT="#000000" LINK="#006666" VLINK="#006666"> <REM --- Checks to see if the instance "reserve" with a value of ---> <REM --- "receipt" is passed to this page. If so, the codes within --- > <REM --- the iCASE and /iCASE tags are executed. This is primarily --- > <REM --- for security purposes as the instanced "reserved" is only --- > <REM --- called in the step immediately before. If a visitor <REM --- accidentally stumbled upon this file, nothing will be run --- > <REM --- unless all previous steps are completed. <iCASE ALIAS="reserve" VALUE="receipt"> **<CENTER>** <TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=5 CELLSPACING=0 WIDTH=500> <TR><TD> <CENTER> **Concert Reservation System** </CENTER> </TD></TR> </TABLE> </CENTER> <REM --- Two steps are combined here: <REM --- 1. The reservation information captured in <REM --- "reserve2.ihtml" is passed to the iPAY tag which

XVII

```
<REM --- processes the information and passes it to CyberCash --->
<REM --- for verification. If successful, the tag returns
<REM --- "success", otherwise it returns a variation of "fail" --- >
<REM --- 2. iIF checks to see if the verification is a success or --- >
<REM --- failure. If successful, it will process the codes</p>
<REM --- immediately after it, which includes updating the
<REM --- database with a successful reservation. If anything --- >
<REM --- other than "success" is returned, ilF will skip to
<REM --- the iELSE tag and execute all codes thereafter.
<iIF COND=<iPAY SERVER="C3"</pre>
              amount=:total
              id=:oid
              ccnum=":ccnum"
              ccmexp=":ccmexp"
              ccyexp=":ccyexp"
              name=":name"
              capture="false"
              HOST="http://cr.cybercash.com/cgi-bin"
              PORT=80
              SECRET="vendorid-26"
              CRYPTOKEY="j1y1o1ohNU1ciTdPF1hsvHFjlpnCpR">>
<CENTER>
<TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR><TD>
<TABLE BGCOLOR='#FFFFFF' BORDER=0 CELLPADDING=5 CELLSPACING=1
WIDTH=500>
<TR><TD>
<FONT FACE="Verdana, Arial" SIZE="-1">
<B>The credit card has been approved and the reservation has been processed.<P>
The following are the Authorization Code and Receipt Number:</B>
```

XVIII

Authorization Code: :i_pay_authnumber

Receipt Number: :i_pay_transactionnumber<P> Customer Name: :name
 Total Amount: \$:total
 Credit Card Number: :ccnum
 Month of Expiration: :ccmexp
 Year of Expiration: :ccyexp
 <REM --- Begin database query to update the "customers" table with customer information captured in "reserve2.ihtml". <Markup language DBNAME=:datasource</p> database="UPDATE customers SET contact=':name', phone=':phone', email=':email' WHERE id=:custid"> <REM --- Begin database query to update the "orders" table is updated with the approval code returned by CyberCash, as well as the transaction information (total charge, credit card number, etc). --- > <Markup language DBNAME=:datasource</pre> database="UPDATE orders SET approvalcode=':i_pay_authnumber', receiptnum=':i_pay_transactionnumber', totalcharge=:total,cc=':ccnum', ccm=':ccmexp', ccy=':ccyexp' WHERE id=:oid" FAILURE="The information was not saved correctly.
"> <REM --- Begin database query to turn off the availability of the seats that have been reserved by setting the "active" attribute of the seats to "0". --- > <database ALIAS="markoff" dbname=":datasource" database="SELECT pid FROM</p> orderdetail WHERE oid=:oid"> <databaseFETCH ALIAS="markoff">

<iWHILE NOTALIAS=i databaseempty>

<Markup language dbname=":datasource"

XIX

database="UPDATE products

SET active=0

WHERE id=:markoff_pid">

<databaseFETCH ALIAS="markoff">

</iWHILE ALIAS=i_databaseempty>

```
</database ALIAS="markoff">
</FONT>
</TD></TR>
</TABLE>
</TD></TR>
</TABLE>
</CENTER>
<iELSE>
<CENTER>
<TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR><TD>
<TABLE BGCOLOR='#FFFFFF' BORDER=0 CELLPADDING=5 CELLSPACING=1
WIDTH=500>
<TR><TD>
<FONT FACE="Verdana,Arial" SIZE="-1"><B>
This transaction could not be processed by Cyber Cash at this time. Either the Credit Card
Information was invalid or the Cyber Cash server is currently not available.
</B></FONT>
</TD></TR>
</TABLE>
</TD></TR>
</TABLE>
</CENTER>
</iIF>
```

<FORM ACTION="index.ihtml" METHOD="post">

</FORM>

```
<CENTER>
<TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR><TD>
<TABLE BGCOLOR='#FFFFFF' BORDER=0 CELLPADDING=5 CELLSPACING=1
WIDTH=500>
<TR><TD>
<CENTER>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>
<INPUT TYPE="submit" VALUE="Return to Concert Selection">
</B></FONT>
</CENTER>
</TD></TR>
</TABLE>
</TD></TR>
</TABLE>
</CENTER>
</FORM>
<FORM ACTION="http://domainname/filename" METHOD="post">
<CENTER>
<TABLE BGCOLOR='#000000' BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR><TD>
<TABLE BGCOLOR='#FFFFFF' BORDER=0 CELLPADDING=5 CELLSPACING=1
WIDTH=500>
<TR><TD>
<CENTER>
<FONT FACE="Verdana, Arial" SIZE="-1"><B>
<INPUT TYPE="submit" VALUE="Return to On-Line Ticketing & Reservation Front Desk">
</B></FONT></CENTER>
</TD></TR></TABLE></TD></TR></CENTER>
```

which verifies and confirms payment information

End Figure 5

XXI

</iCASE ALIAS="reserve"> </BODY>